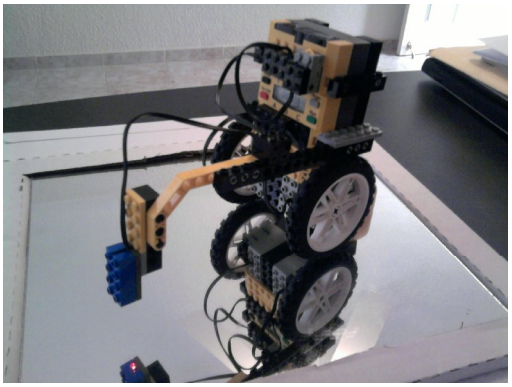
SISTEMAS DE CONTROL I

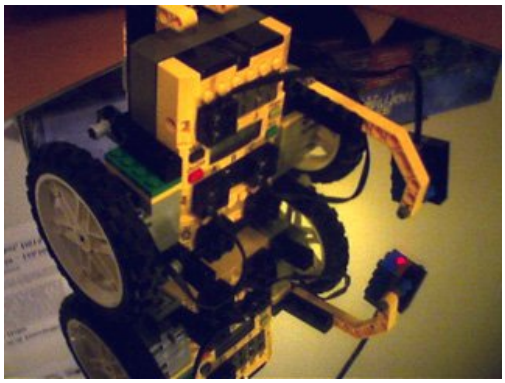
PROYECTO FINAL

CONSTRUCCIÓN Y CONTROL DE UN ROBOT AUTO-BALANCEADO UTILIZANDO LEGO MINDSTORM RCX

- **OBJETIVO:**

  Construir un robot auto-balanceado en su posición de operación inestable (péndulo invertido) con base a un sistema educativo Lego Mindstorm RCX, el cual logre este objetivo por medio de accionar los motores en sus 2 ruedas y utilizar un sensor infrarrojo para detectar la inclinación del robot; además el robot debe detenerse por un sensor de límite. El algoritmo de control será programado por medio del lenguaje RobotC y debe mantener al robot balanceado al menos por 15 segundos.





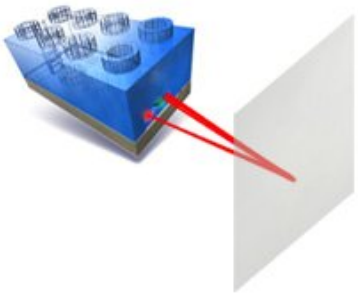
- **REGLAS DE OPERACIÓN:**

  Los alumnos trabajarán en el Laboratorio de Instrumentación y Control en equipos de 4 o 5 alumnos, donde a cada equipo se le asignará un kit educativo Lego Mindstorm RCX y una PC para la programación del robot. **Aunque debido al espacio limitado del laboratorio, solo 2 miembros del equipo pueden estar en el laboratorio al mismo tiempo**. Cada computadora tiene instalado el software RobotC (http://www.robotc.net), y una carpeta en el escritorio que contiene un manual de usuario con prácticas para facilitar el aprendizaje del programa y una guía de referencia. El laboratorio se encontrará abierto de las 8:00-14:00 hrs y de las 17:00-20:00 hrs. a partir del 5/mayo hasta el 21/mayo. **Cada equipo debe hacerse responsable de cuidar el material que se le haya asignado, así como no instalar ningún otro software en las computadoras. Además los equipos**

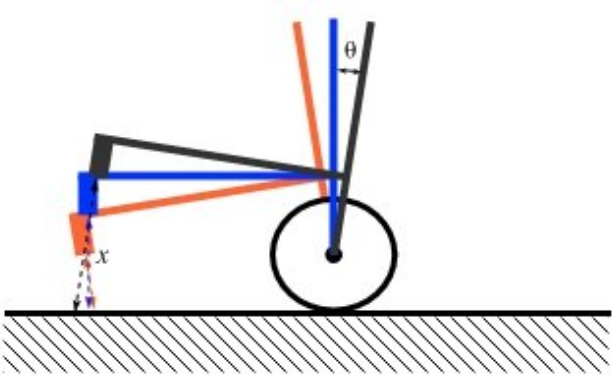
deberán ser respetuosos del trabajo y material de los demás, así como mantener un ambiente adecuado de trabajo en el laboratorio. No se permitirán la entrada a estudiantes ajenos a la materia de Sistemas de Control I al Laboratorio de Instrumentación y Control para propia seguridad de los equipos. Después de la evaluación experimental, cada equipo deberá entregar el kit educativo Lego Mindstorm RCX tal como fue recibido.

- **<u>DESCRIPCIÓN DE ACTIVIDADES:</u>**

Primeramente los integrantes del equipo deberán construir su robot según su propio ingenio, y programarlo para realizar el balanceo del mismo, donde para el diseño tendrán en cuenta el peso resultante y una distribución uniforme. Por otro lado, la medición que permitirá evaluar si el robot se encuentra en la posición de equilibrio, será el sensor infrarrojo, que se encuentra formado por un fotodiodo y un fototransistor, es decir con base a la luz emitida por el fotodiodo, el fototransistor detectará una intensidad de luz resultante:



La intensidad de esta luz incidente puede correlacionarse con la distancia a la superficie de rodamiento *x*, y esta a su vez con el desplazamiento angular θ, utilizando la siguiente configuración:



Por lo tanto, *si la medición de intensidad se incrementa significa que el robot se inclina hacia el enfrente* y en consecuencia el robot deberá moverse hacia atrás. Mientras tanto, *si el medición se decrementa significa que el robot se inclina hacia atrás*, y el robot debe accionarse al frente. En consecuencia, dentro del algoritmo de balanceo, primeramente se debe programar una rutina para definir la posición de equilibrio y guardarse esta medición como referencia del algoritmo de control. Enseguida con base al error de regulación (error= referencia – medición) se sugiere utilizar una ley de control PD (proporcional-derivativa) para accionar los motores del robot. Cabe mencionar que se sugiere utilizar un

espejo como superficie de rodamiento, de forma que exista un reflejo uniforme de la luz proveniente del fotodiodo en todo momento.

La plataforma de programación será RobotC, que es un lenguaje basado en C para implementar algoritmos en el Lego Mindstorm RCX y NXT (http://www.robotc.net). La comunicación entre el robot Lego y la computadora se realiza por medio de una Torre de Transmisión Infrarroja, para enviar y recibir información de forma inalámbrica, aunque debe cuidarse siempre mantener una línea de vista entre el robot y la torre. Este lenguaje tiene funciones especiales para leer los puertos de medición y activar los motores, aunque deben cuidarse ciertos aspectos:

- La implementación se realiza en tiempo discreto, por lo que el control PD estaría dado por

  Accionamiento del motor (k) = - Kp * error(k) - Kd*[error(k)-error(k-1)]
- Los cálculos en la implementación se realizan con números enteros,
- Los motores no tienen una relación uniforme entre el voltaje aplicado y revoluciones de la flecha (se sugieren ganancias de control diferentes por cada motor),
- Las mediciones del sensor infrarrojo contienen ruido, por lo que sugiere implementar un promediador de 2º orden → medición = (medición + lectura del sensor)/2.
- En el calculo del error puede incluirse una zona muerta para evitar un accionamiento continuo de los motores

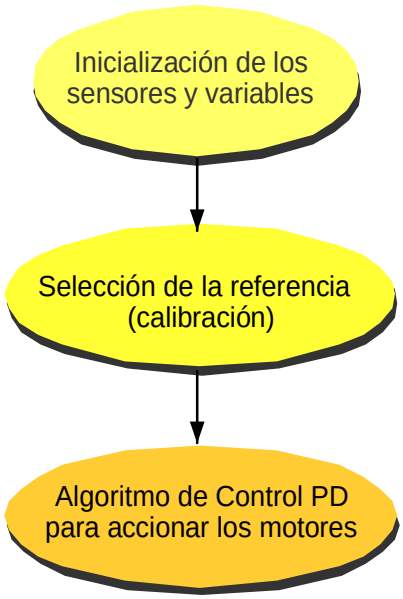
  Si |error(k)| >= umbral

  Accionamiento del motor → Ley de Control PD

  Contrario

  Accionamiento del motor → 0

La estructura global del algoritmo de control estaría dada por


Inicialización de los sensores y variables
Selección de la referencia (calibración)
Algoritmo de Control PD para accionar los motores

La evaluación del algoritmo de control se realizará dependiendo del tiempo que el robot se mantenga balanceado, y el equipo de logre el mayor tiempo con base a 3 mediciones será el ganador. La puntuación de la evaluación experimental será como sigue:

Alcance	Puntuación
Construcción apropiada y diseño robusto	60 pts.
Estabilización entre 5 y 10 segs	80 pts
Estabilización entre 10 y 15 segs	90 pts
Estabilización >= 15 segs	100 pts.

El equipo ganador y que cumpla con un tiempo mayor a 15 segs obtendrá un punto extra sobre la calificación final.

Finalmente, después de la evaluación, cada equipo deberá entregar un reporte donde incluyan fotos de su robot, el o los programas implementados, el registro del tiempo de estabilización y conclusiones del proyecto. La calificación del proyecto se conformará por el resultado de la evaluación experimental (50%) y la calidad del reporte (50%).

- **ETAPAS**:

El desarrollo de los objetivos del proyecto puede llevarse a cabo a partir de 6 etapas clave:

I. Construcción del Robot
II. Prácticas de prueba en el software RobotC
III. Desarrollo del algoritmo de balanceo
IV. Implementación del algoritmo de balanceo y depuración.
V. Evaluación experimental y presentación del algoritmo de control → **lunes 24/mayo 12 hrs (Sala de Usos Múltiples).**
VI. Elaboración y entrega del reporte final → **martes 25/mayo**

- **EQUIPOS:**

EQUIPO #1	**EQUIPO #2**	**EQUIPO #3**	**EQUIPO #4**
Diana Hernández (C)	**Daniel Viramontes (C)**	**Oscar Zamarripa (C)**	**Víctor Hugo De León (C)**
José Camacho	Alberto Reyna	Josue Natera	Fco. Javier Padrón
Fernando Estrada	Rodrigo Azuara	Fco. Gerardo Gutierrez	Salvador García
Paulina Lepez	José Carmen López	Juan Carlos Navarro	Elizabeth Morin
	Joaquín Cantú	Orlando Correa	